```python
from a0_items import *
import pandas


################################################################################
##############################
### Read CSV
################################################################################
##############################

#read csv file by list index
def CSV_reader (path_CSV):

    LIST_of_list = []

    with open(path_CSV, 'r', encoding="ascii", errors="ignore") as file:

        csvreader = csv.reader(file,delimiter=':')

        for row in csvreader:

            LIST_of_list.append(row)

    return LIST_of_list


'''
for x in CSV_reader (path_CSV='../../data/raw/RAW_country_language/cities.csv'):

    for y in x:

        y_list = y.split(',') #must convert string back to list first

        print (y_list[1])
'''


# create & read csv file by column name
def CSV_dict_reader (path_CSV):

    LIST_data = []

    with open(path_CSV, 'r', encoding="ascii", errors="ignore") as file:

        csvreader = csv.DictReader(file)

        for row in csvreader:

            LIST_data.append(row)

    return LIST_data


'''
for x in CSV_dict_reader (path_CSV='../../data/raw/RAW_country_language/cities.csv'):

    print (x['name'])
'''


################################################################################
##############################
### create CSV header
################################################################################
##############################
```

```python
64
65      #https://adamtheautomator.com/read-csv-in-python/
66      def CSV_create_header_1 (path_CSV, LIST_header):
67
68          if os.path.exists (path_CSV):
69
70              print (path_CSV + ' Exists')
71
72          else:
73
74              print (path_CSV + ' NOT exists, is created')
75
76              with open(path_CSV, 'w', newline='') as outcsv:
77
78                  writer = csv.writer(outcsv)
79
80                  writer.writerow(LIST_header)
81
82      '''
83      header_LIST = ["Date", "temperature 1", "Temperature 2"]
84      CSV_create_header_1 (path_CSV='test_write_header.csv', LIST_header=header_LIST)
85      '''
86
87
88      def CSV_create_header_2 (path_CSV, LIST_header):
89
90          # read contents of csv file
91          file = pandas.read_csv(path_CSV)
92          print("\n Original file:")
93          print(file)
94
95          # converting data frame to csv
96          file.to_csv(path_CSV, header=LIST_header, index=False)
97
98          # display modified csv file
99          file2 = pandas.read_csv(path_CSV)
100         print('\n Modified file:')
101         print(file2)
102
103
104     #headerList = ['name', 'number', 'skills']
105     #CSV_create_header_2 (path_CSV='test_panda.csv', LIST_header=headerList)
106
107
108     ################################################################################
109     ### create CSV data
110     ################################################################################
111
112     #https://adamtheautomator.com/read-csv-in-python/
113     def CSV_create_data_1 (path_CSV, LIST_data):
114
115         if os.path.exists (path_CSV):
116
117             print (path_CSV + ' Exists')
118
119         else:
120
121             print (path_CSV + ' NOT exists, is created')
122
123             with open(path_CSV, 'a', newline='') as outcsv: #use 'a' to allow append of data
                 by looping
124
125                 writer = csv.writer(outcsv)
126
127                 writer.writerow(LIST_data)
```

```python
128
129     '''
130     data_LIST = ["data1", "data2", "data3"]
131     CSV_create_header_1 (path_CSV='test_write_header.csv', LIST_header=header_LIST,
        LIST_data=data_LIST)
132     '''
133
134
135
136     #List should be created separated to loop through this function
137     def CSV_update_data_2 (path_CSV, col_index, col_name, col_value):
138
139         # reading the csv file
140         df = pandas.read_csv(path_CSV)
141
142         # updating the column value/data
143         # col_index starts from 0, its the index of the CSV row and must be an integrer
144         df.loc[col_index, col_name.strip()] = str(col_value)
145
146         # writing into the file
147         df.to_csv(path_CSV, index=False)
148
149         print(df)
150
151
152     #csv_update_data_2 (path_CSV='test_panda.csv', col_index=2, col_name='Col_2',
        col_value='brad')
153
154
155
156     def CSV_update_by_replace (path_CSV, col_name, col_value, replace_with_value):
157
158         # reading the csv file
159         df = pandas.read_csv(path_CSV)
160
161         # updating the column value/data
162         df[col_name] = df[col_name].replace({col_value : replace_with_value})
163
164         # writing into the file
165         df.to_csv(path_CSV, index=False)
166
167         print(df)
168
169
170     #csv_update_by_replace (path_CSV='test_panda.csv', col_name='', col_value='',
        replace_with_value='')
171
```